



PES UNIVERSITY, BANGALORE  
Department of Computer Science and Engineering  
B. Tech (CSE) – 5th Semester – Aug-Dec 2023

---

**UE21CS341A - Software Engineering**  
**PROJECT PLAN DOCUMENT**

*RECIPE RECOMMENDATION SYSTEM*

Shreya Sridhar	PES1UG21CS578	Shubha Masti	PES1UG21CS589
Shriansh Mohanty	PES1UG21CS584	Shyam Krishna	PES1UG21CS935

## Life Cycle Followed

Life-cycle followed: Agile (Scrum Framework)

1. Iterative Development: Agile allows for developing in small, manageable chunks, making errors easier to spot and fix.
2. Feedback Loop: Scrum encourages regular feedback, ensuring the system aligns with user needs.
3. Flexibility: Agile is adaptable to changes, a must for our dynamic web application project.
4. Collaboration: Scrum promotes teamwork, ensuring seamless integration of all project aspects.
5. Quick Launch: Scrum can deliver a basic version quickly, with improvements added in subsequent phases.

By using Agile with Scrum, we can effectively create a user-friendly Recipe Recommendation System, adjusting to feedback and changes as we go.

## Tools Used for this Project

Planning Tools:

- Jira: Advanced project management tool for Scrum, facilitating sprint planning, tracking, and retrospectives.

Design Tools:

- Figma: For creating interactive UI/UX designs and prototypes. Enables collaboration and provides a shared space for design iterations.
- ERDplus: For creating flowcharts, ER diagrams, and system architecture blueprints.

Version Control:

- Git: Widely-used version control system, allowing multiple developers to work simultaneously without overwriting each other's changes.
- GitHub: A cloud-based platform using Git for code storage, pull requests, and code reviews.

Development Tools:

- Visual Studio Code: Popular source code editor with extensions for Python, React, and Flask development.
- Postman: For testing and documenting API endpoints.

Bug Tracking:

- Jira's Bug Tracking: Used in conjunction with its project management capabilities, allows for seamless bug reporting, tracking, and resolution.

Testing Tools:

- pytest: A framework for Python unit tests, ensuring the backend logic works as expected.
- jest: JavaScript testing framework, mainly used for testing React applications.

## **Deliverables (classified as reuse/build components)**

### User Authentication System

- Category: Reuse
- Justification: Many projects require user authentication. Leveraging existing libraries or frameworks like Flask-Login or OAuth can expedite the process and ensure security without reinventing the wheel.

### Recipe Recommendation Algorithm

- Category: Build
- Justification: The core logic for recommending recipes based on available ingredients and user preferences is unique to this project and requires a custom build.

### User Interface (UI) Components

- Category: Reuse
- Justification: Many standard UI components, like buttons, input fields, or modals, can be reused from libraries like React-Bootstrap or Material-UI to ensure consistency and save development time.

### Bookmark Feature for User Favourites

- Category: Build
- Justification: Custom integration is needed to align bookmarking with our unique user profiles and recipe database, ensuring easy saving and retrieval of favourites.

### Database Schema for Recipe Storage

- Category: Build
- Justification: The way recipes, ingredients, and user data interrelate is specific to this project. A custom schema will be necessary to cater to these unique relationships and data storage needs.

### Advanced Filtering Logic (based on kcal, allergies, etc.):

- Category: Build
- Justification: The specific filters, especially those related to dietary restrictions or preferences, will likely require a custom build to ensure accuracy and user satisfaction.

## Work Breakdown Structure

### 1. Recipe Recommendation System Development

#### 1.1. Requirement Analysis

- 1.1.1. Gather requirements
- 1.1.2. Prioritise features
- 1.1.3. Define system constraints

#### 1.2. System Design

- 1.2.1. Frontend design
- 1.2.2. Backend architecture design
- 1.2.3. Database schema design

#### 1.3. Development

- 1.3.1. Frontend Development
  - 1.3.1.1. User Interface development
  - 1.3.1.2. React components for features
- 1.3.2. Backend Development
  - 1.3.2.1. Flask application setup
- 1.3.3. Database Development
  - 1.3.3.1. Database setup and connection
  - 1.3.3.2. Data insertion and retrieval logic

#### 1.4. Testing

- 1.4.1. Unit Testing
- 1.4.2. Integration Testing
- 1.4.3. User Acceptance Testing (UAT)

#### 1.5. Deployment

- 1.5.1. Set up hosting environment
- 1.5.2. Deploy frontend and backend

#### 1.6. Maintenance and Support

- 1.6.1. Monitor system performance
- 1.6.2. Address issues
- 1.6.3. Periodic updates

Effort Estimation (in person-months)

Task	Days	Person-months
Database Design	5	0.23
System Design	9	0.414
Frontend Design	6	0.276
Application pipeline	5	0.23
Integration	5	0.23
Fullstack testing	4	0.184
Debugging	5	0.23
Project Performance testing	5	0.23
Deployment	2	0.092

Gantt Chart

GANTT CHART

PROJECT TITLE - RECIPE RECOMMENDATION SYSTEM



